Mastermath midterm examination
Parallel Algorithms. Retake.

Teacher: Rob H. Bisseling, Utrecht University

December 19, 2018

Each of the four questions is worth 10 points. Total time 120 minutes. Motivate you answers!

1. Let $\mathbf{x}$ be a vector of length $n \geq 1$.
   (a) [3 pt] Define the *cyclic distribution* of $\mathbf{x}$ over p processors.
   (b) [3 pt] Define the *block distribution* of $\mathbf{x}$ over p processors.
   (c) [2 pt] Give an example of a vector operation where the block distribution is preferred over the cyclic distribution.
   (d) [2 pt] Analyse its BSP cost for both distributions.

2. Let f be a polynomial of degree $n-1$ in a real variable x, which is given by its coefficients $a_0, a_1, \ldots, a_{n-1}$, such that

$$f(x) = \sum_{i=0}^{n-1} a_i x^i.$$

   Assume that the coefficients are distributed over the p processors of a parallel computer by the block distribution, where $n \bmod p = 0$.

   (a) [5 pt] Design an efficient parallel algorithm for the evaluation of this polynomial for a given value of x, where on input x is only available in processor $P(0)$. The output $f(x)$ only needs to become available at $P(0)$. Formulate the algorithm in the notation we have learned, with pseudocode for processor $P(s)$, where $0 \leq s < p$.
   (b) [5 pt] Analyse the BSP cost of your algorithm.

3. Let A be an $n \times n$ dense matrix, distributed by the $q \times q$ square cyclic distribution over $p = q^2$ processors, with $n \bmod q = 0$. We want to permute all the rows of the matrix, such that row i moves to the position of row $(i+1) \bmod n$.

   (a) [5 pt] Design an algorithm for this permutation of matrix A. Formulate the algorithm in the notation we have learned, with pseudocode for processor $P(s,t)$, where $0 \leq s, t < q$.

   (b) [5 pt] Analyse the BSP cost of your algorithm.

4. Let A be an $n \times n$ dense matrix, distributed by the $q \times q$ square block distribution $\phi = (\phi_0, \phi_1)$ over $p = q^2$ processors, with $n \bmod p = 0$. This means that matrix element a_{ij} is assigned to processor $P(\phi_0(i), \phi_1(j))$. Let $\mathbf{x}$ be a vector of length n. We want to perform the symmetric rank-1 update

$$A := A + \mathbf{x}\mathbf{x}^{\mathrm{T}}.$$

   (a) [4 pt] Choose a suitable distribution $\phi_{\mathbf{x}} = (\phi_{\mathbf{x},0}, \phi_{\mathbf{x},1})$ for the vector $\mathbf{x}$ that will facilitate the desired operation. This means that vector element x_i is assigned to processor $P(\phi_{\mathbf{x},0}(i), \phi_{\mathbf{x},1}(i))$. Note that it must be a true distribution, so every vector component x_i is owned by exactly one processor. Give the reason(s) for your choice.

   (b) [3 pt] Design an efficient algorithm for the parallel rank-1 update for the distributions ϕ and $\phi_{\mathbf{x}}$. You may describe the algorithm in words, or in pseudocode in the notation we learned.

   (c) [3 pt] Analyse the BSP cost of your algorithm